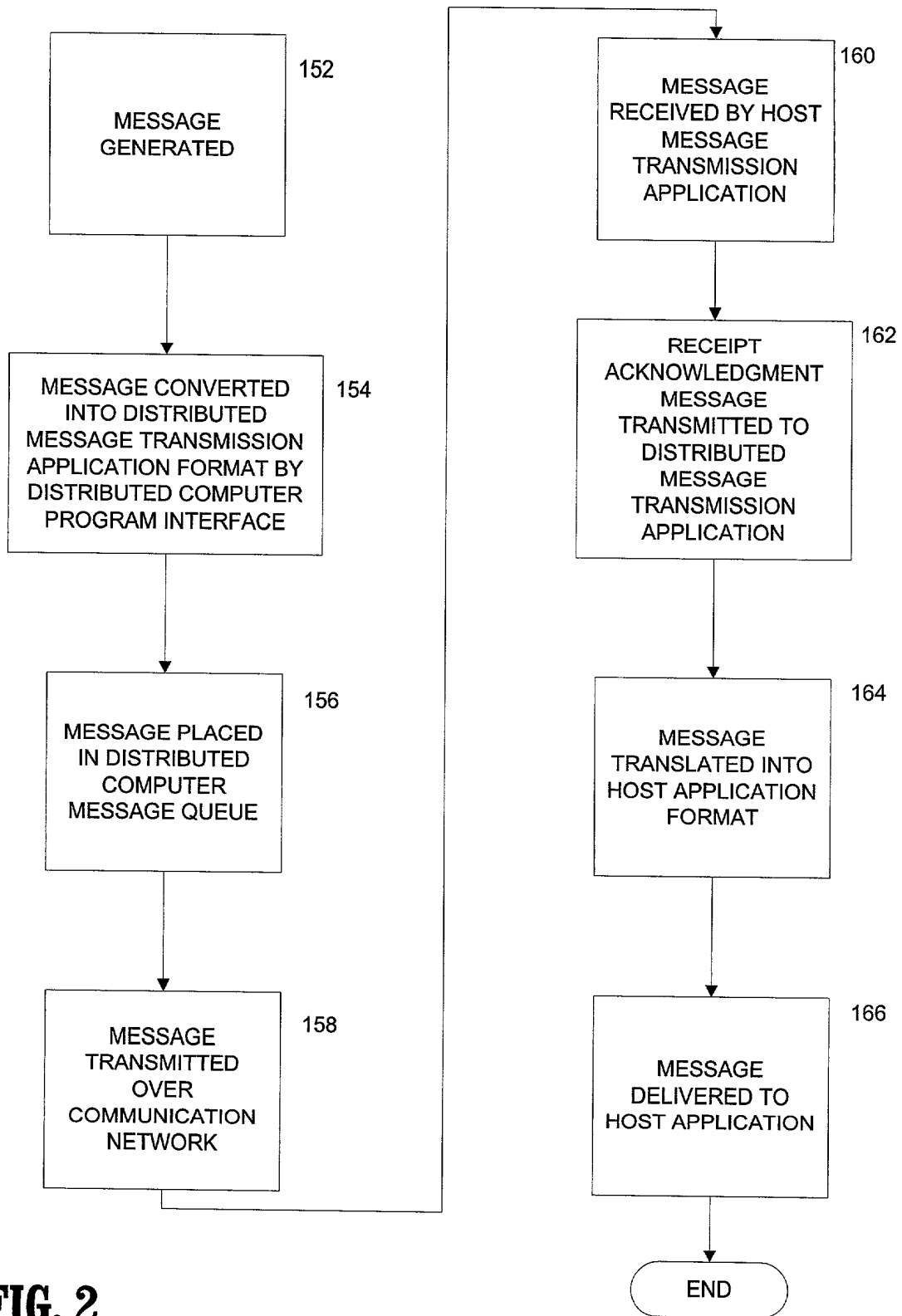


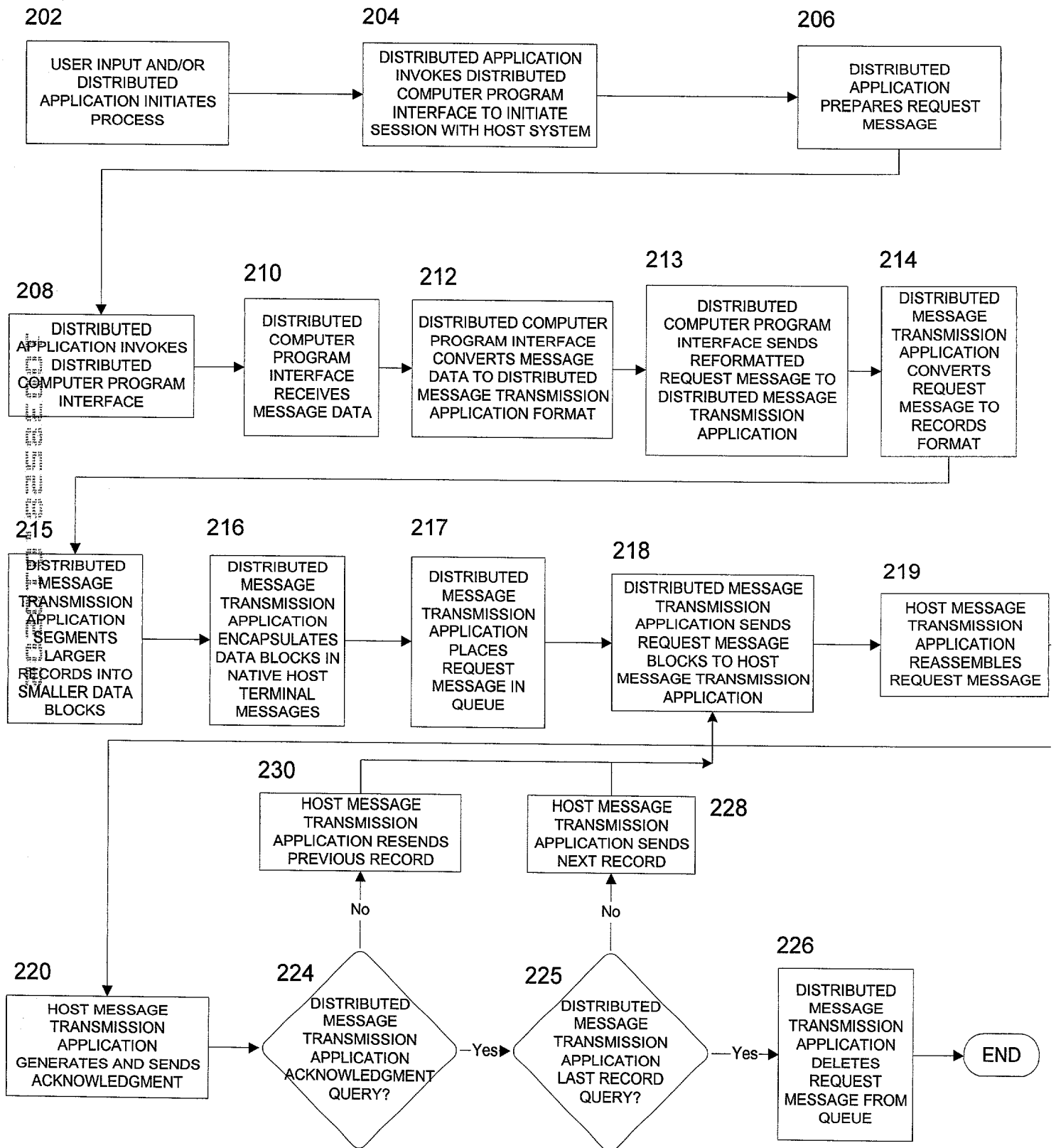
**Fig. 1**

150



**FIG. 2**

200



**FIG. 3**

FIG. 4A is a block diagram of a file 300. The file 300 is composed of a sequence of records, including Record 1, Record 2, Record 3, Record 4, and Record N. A double-headed arrow indicates the sequence of records. A curved arrow labeled 302a-n points from the sequence of records to the detailed structure of a single record shown in FIG. 4B.

300

File



302a-n

FIG. 4A

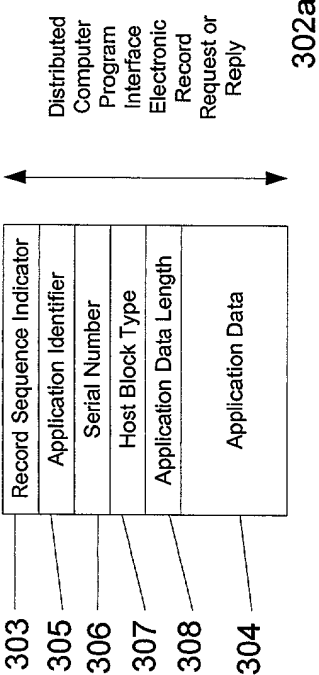


FIG. 4B

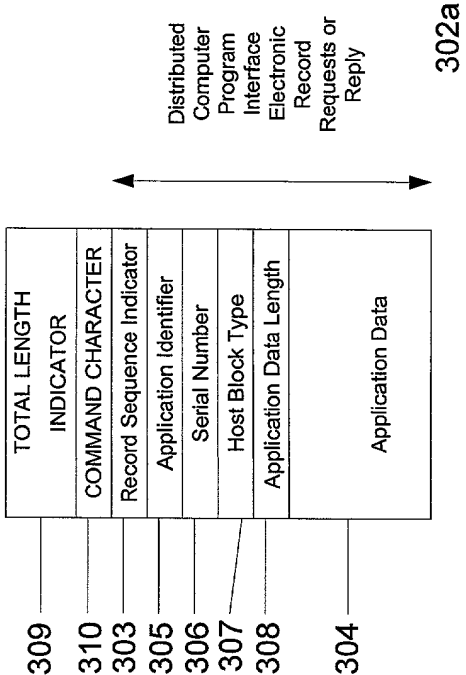
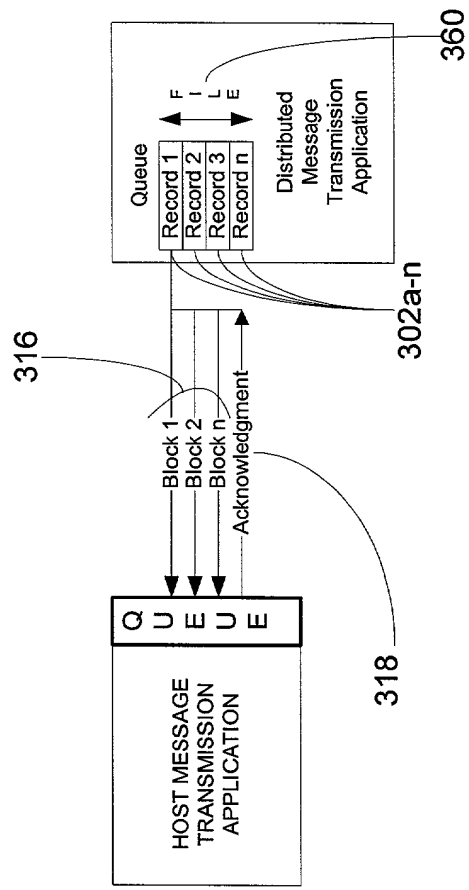
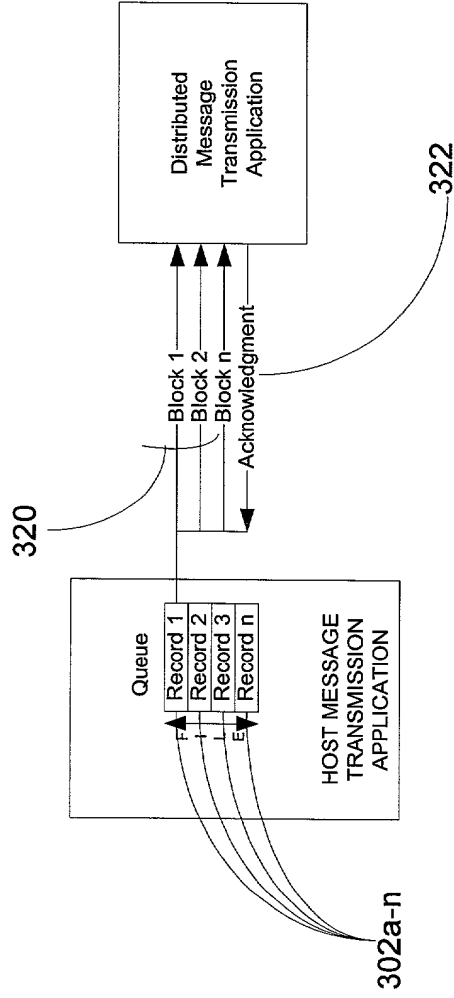


FIG. 4C



**FIG. 5**

FIG. 6 is a block diagram of a system for distributed message transmission. The system includes a Host Message Transmission Application (302a-n) and a Distributed Message Transmission Application (322). The Host Message Transmission Application (302a-n) includes a Queue (320) and a set of records (Record 1, Record 2, Record 3, Record n). The Queue (320) is connected to the Distributed Message Transmission Application (322) via a set of lines (302a-n). The Distributed Message Transmission Application (322) includes a set of blocks (Block 1, Block 2, Block n) and an Acknowledgment block. The Queue (320) sends data to the Distributed Message Transmission Application (322) via the set of lines (302a-n). The Distributed Message Transmission Application (322) sends an Acknowledgment back to the Queue (320).



**FIG. 6**

FIG. 7 is a block diagram of a message format 330. The message format 330 is divided into three sections: a Defined Format Message Indicator 332, a Record Data 334, and an End of Message Character 336. The message format 330 is used for distributed message transmission, application data block request or reply.

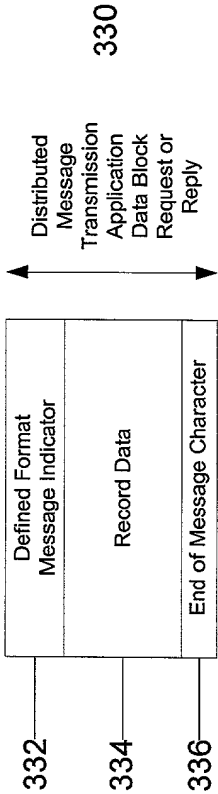


FIG. 7

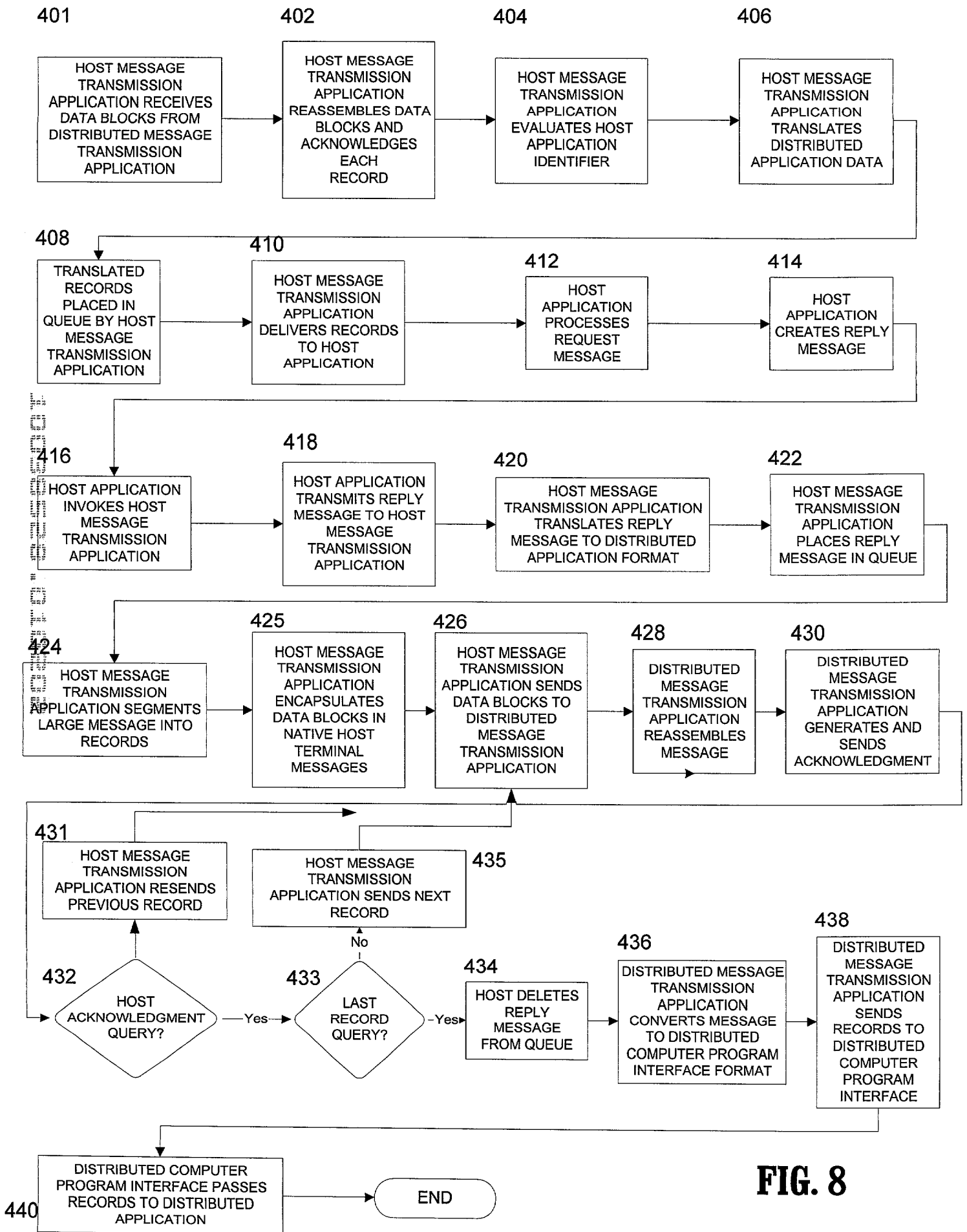


FIG. 8



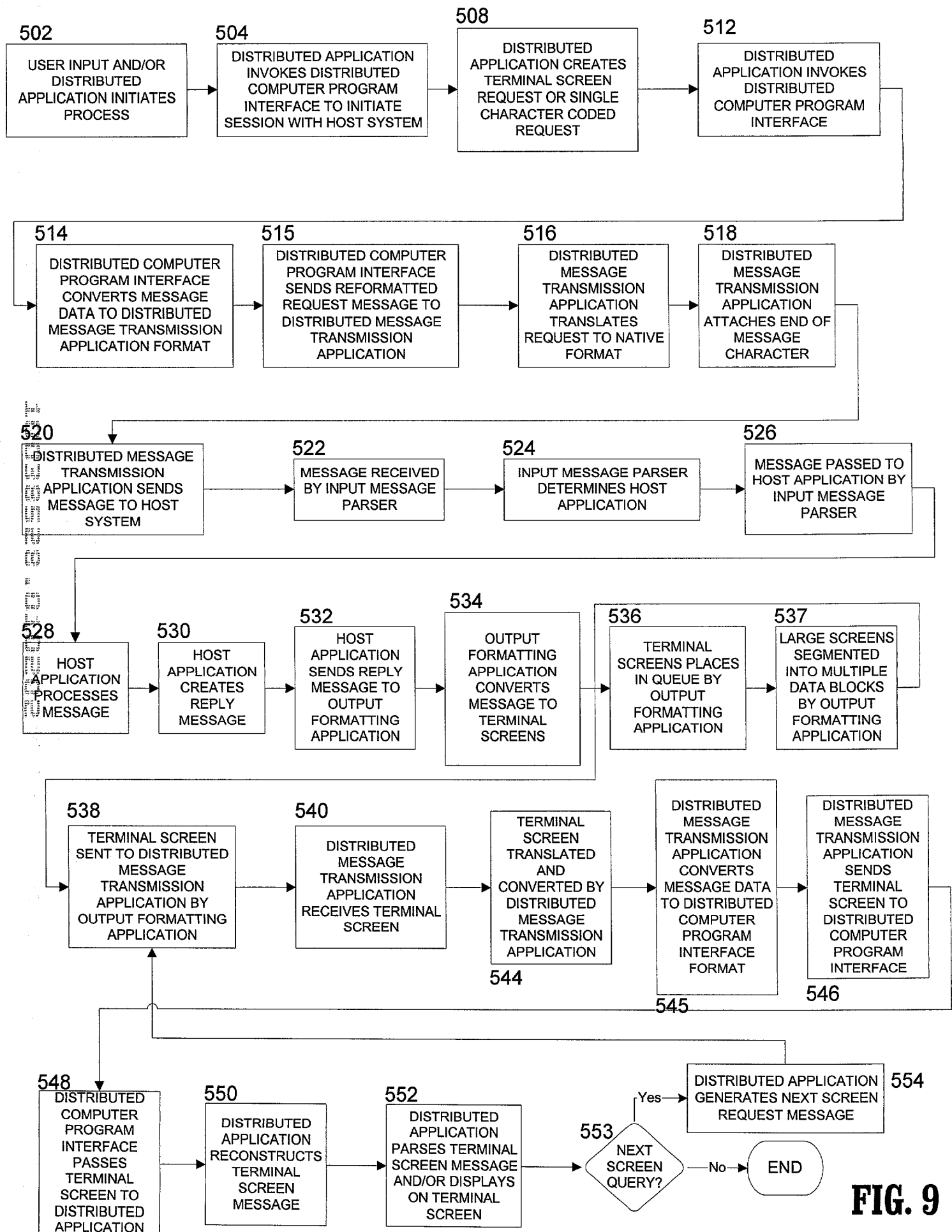


FIG. 9

FIG. 10 is a block diagram of a terminal screen reply message 600. The message 600 is composed of a plurality of blocks, including Block 1, Block 2, Block 3, and Block n. The blocks are arranged in a sequence, and the message 600 is labeled as 600a-n.

600

Terminal Screen Reply Message

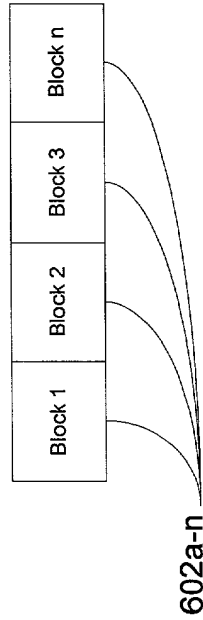
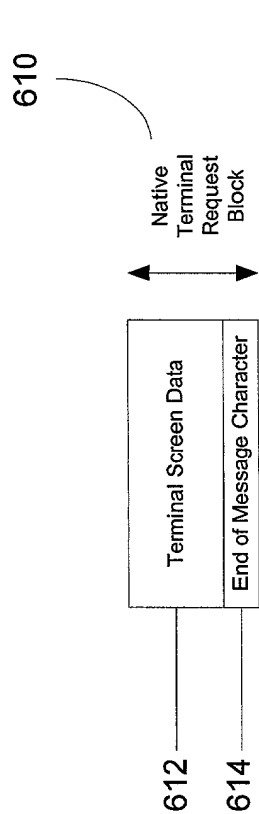
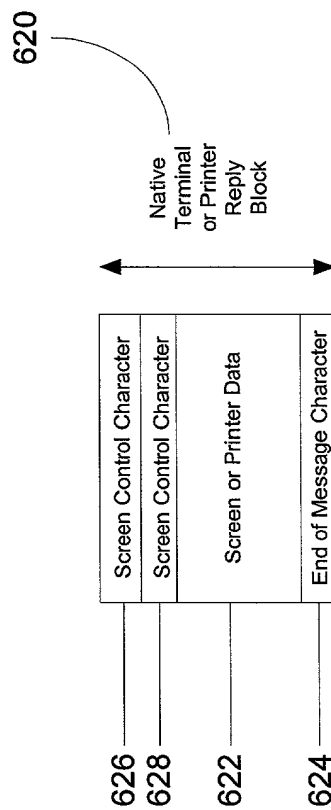


FIG. 10

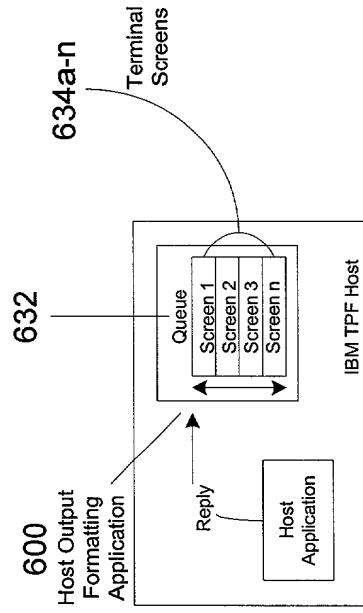


**FIG. 11A**



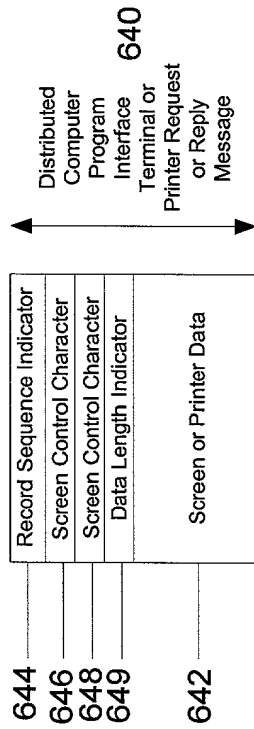
**FIG. 11B**

IBM CORPORATION  
ARMONK, NEW YORK 10504  
U.S. PATENT OFFICE

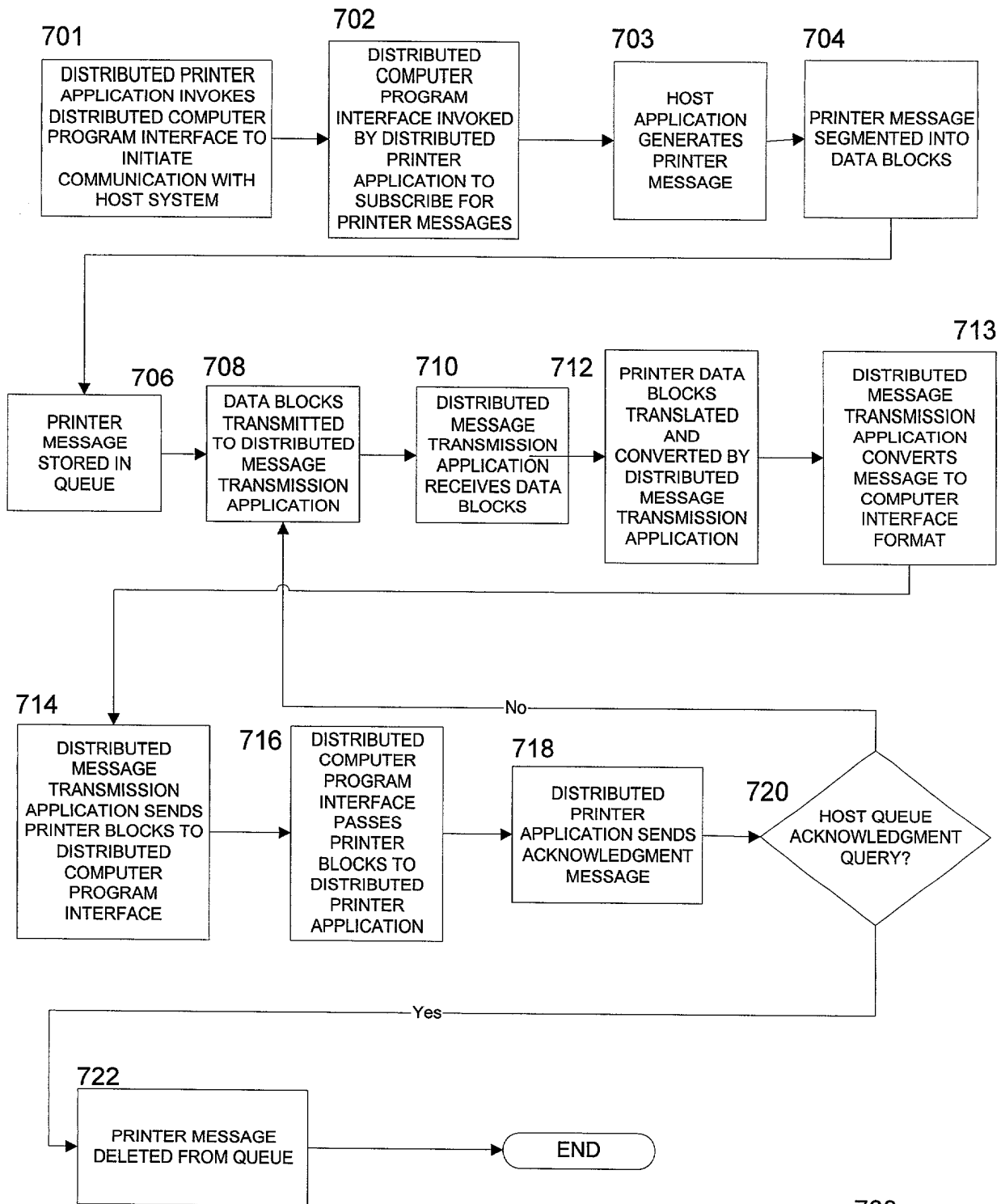


**FIG. 12**

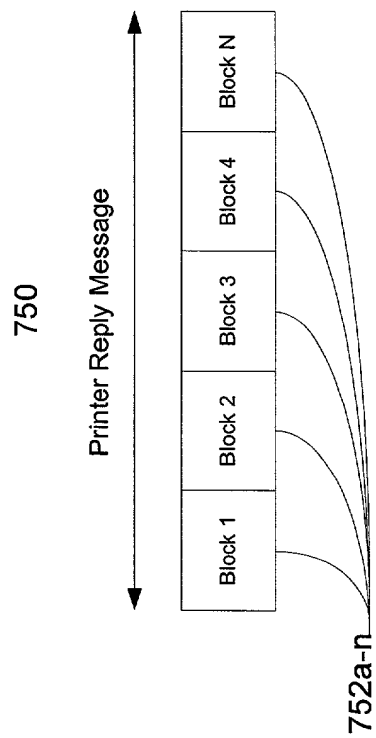
FIG. 13 is a block diagram of a data structure for a distributed computer program interface terminal or printer request or reply message. The data structure is shown as a rectangular box divided into four horizontal sections. The top section is labeled "Record Sequence Indicator" and is connected to reference numeral 644. The second section is labeled "Screen Control Character" and is connected to reference numeral 646. The third section is labeled "Screen Control Character Data Length Indicator" and is connected to reference numeral 648. The bottom section is labeled "Screen or Printer Data" and is connected to reference numeral 642. A double-headed arrow points from the data structure to the text "Distributed Computer Program Interface Terminal or Printer Request or Reply Message 640".



**FIG. 13**

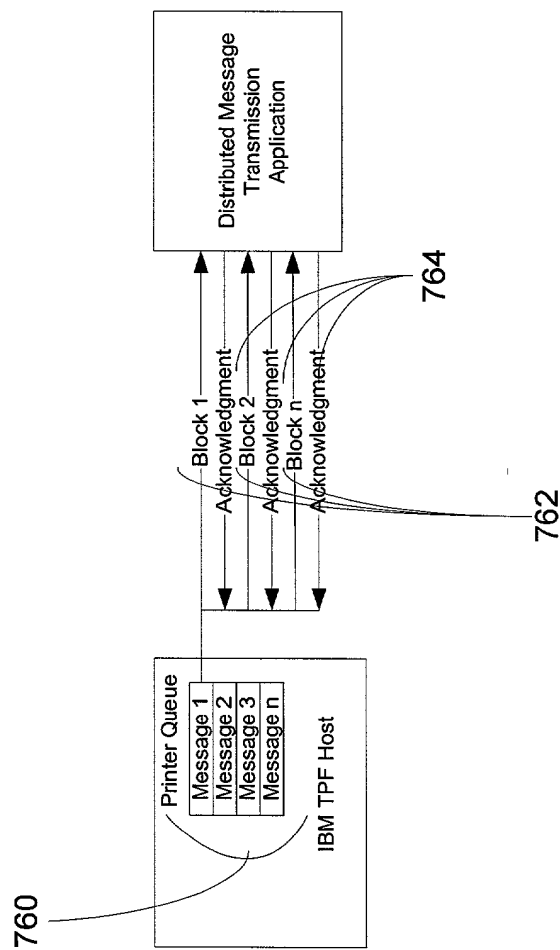


**FIG. 14**

[illegible]

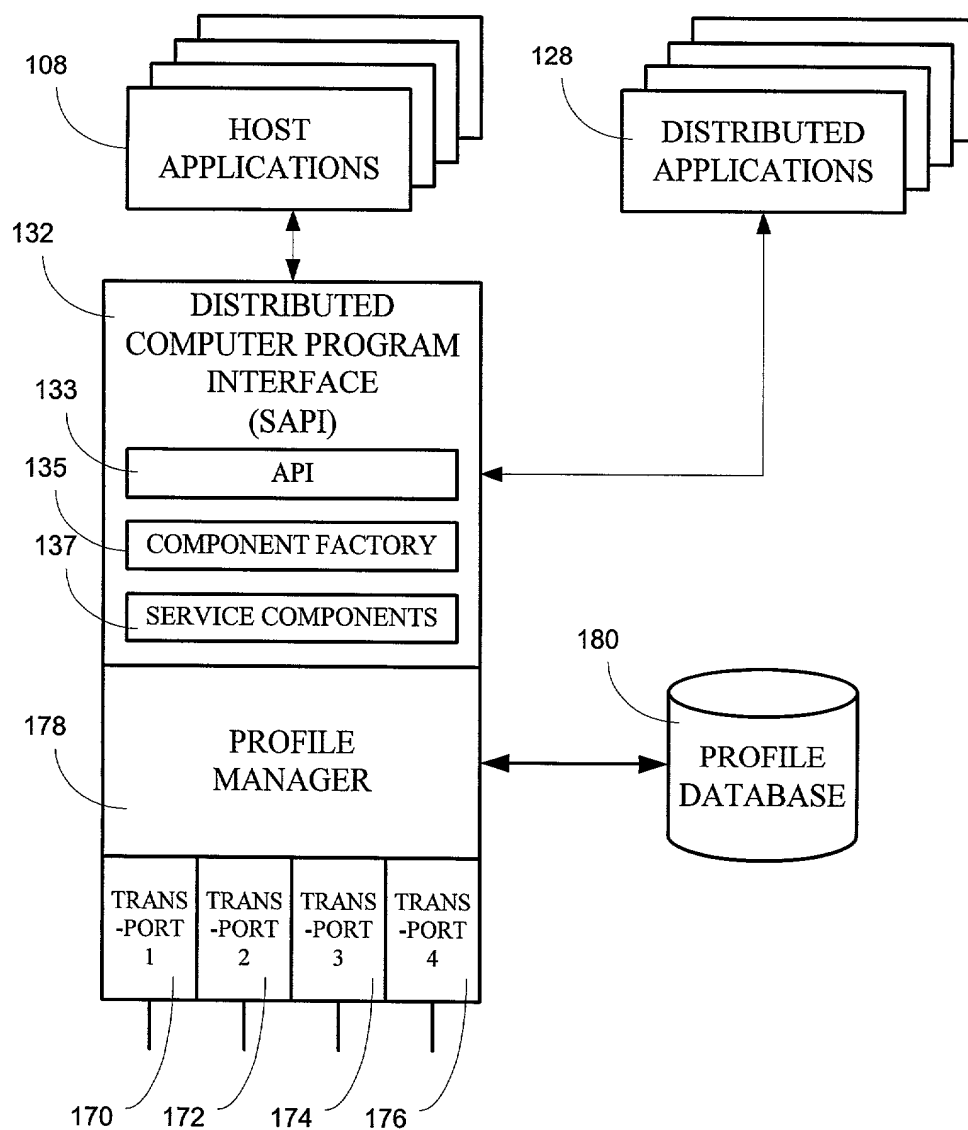
**FIG. 15**

755

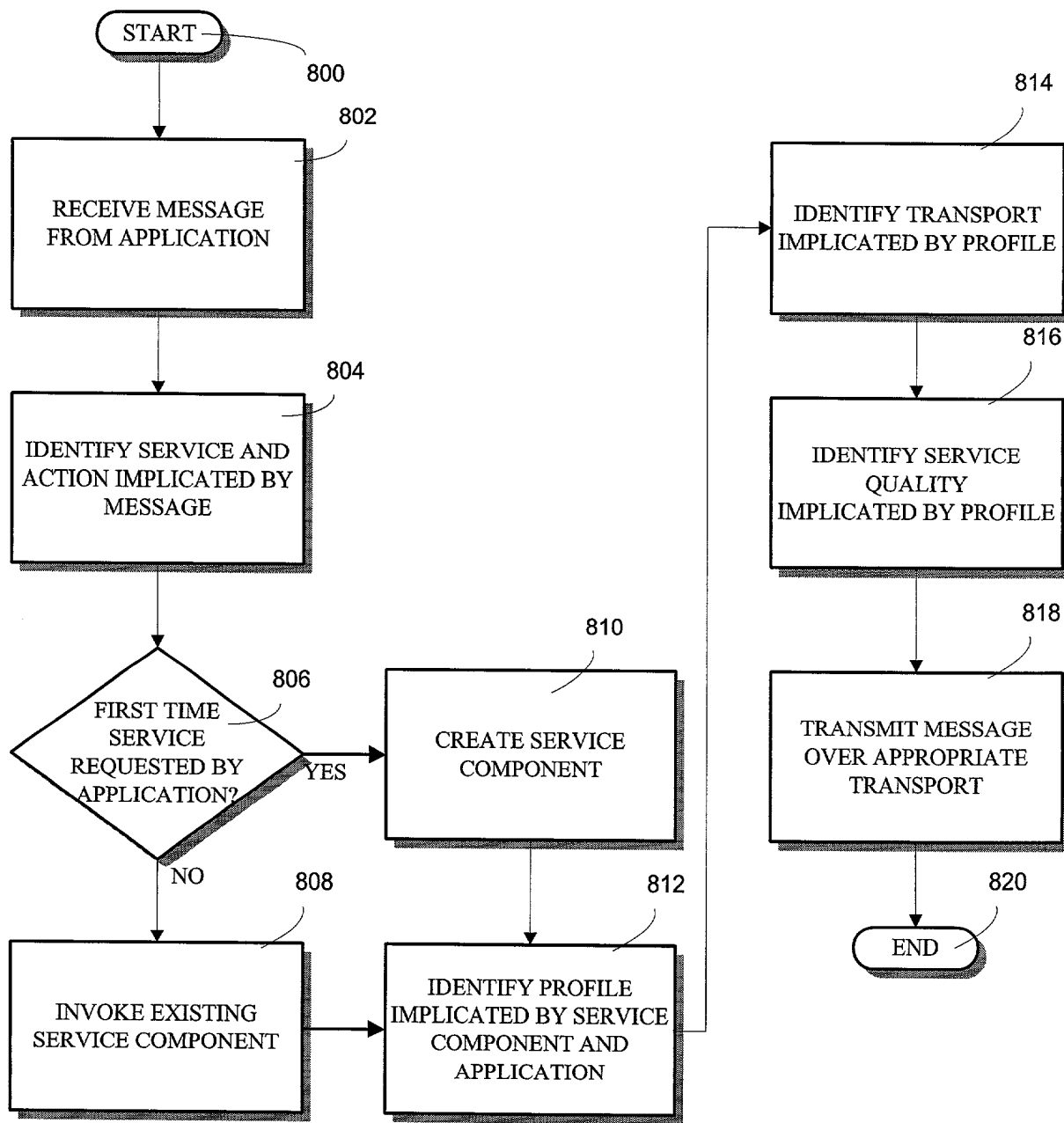


**FIG. 16**

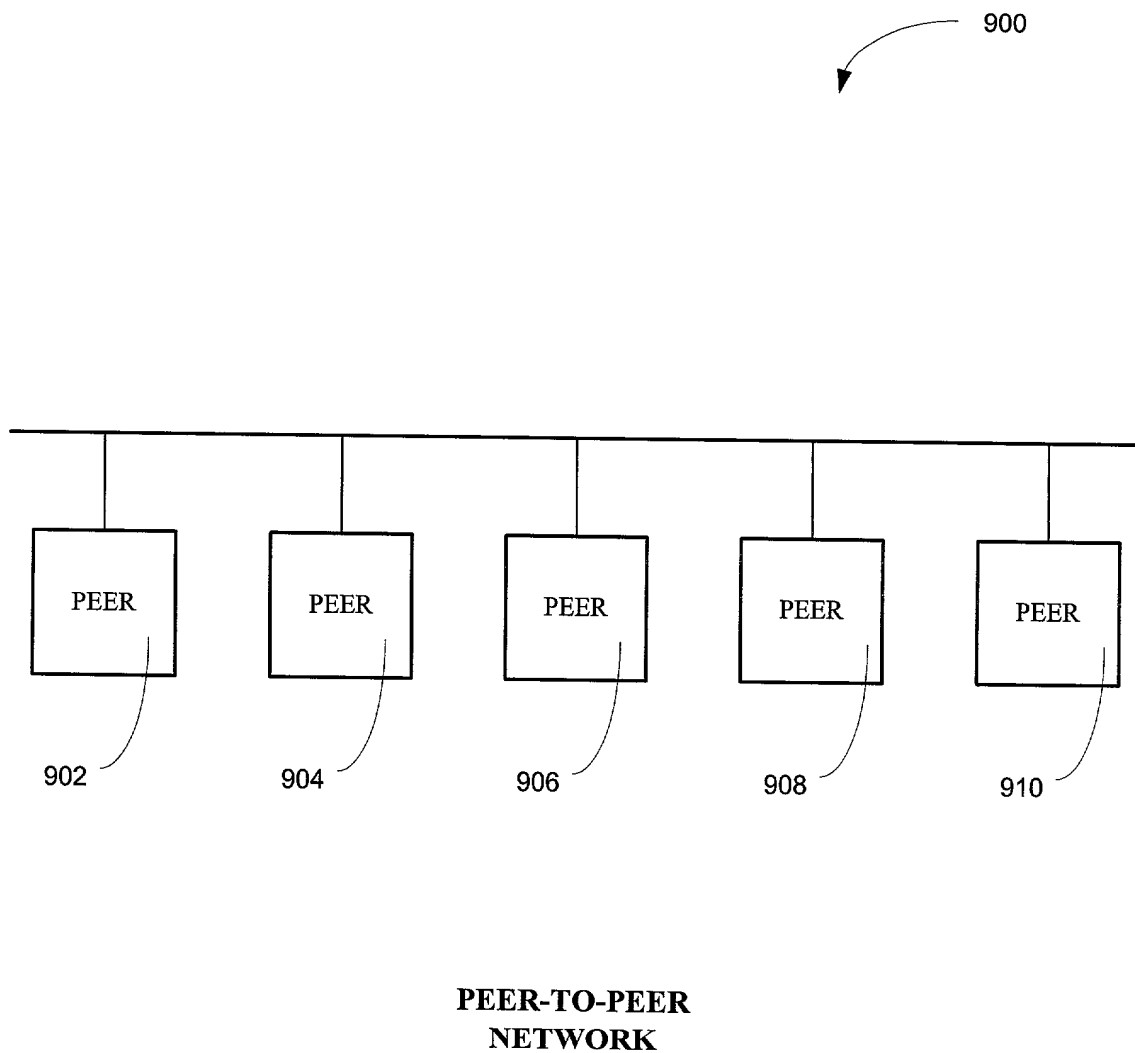




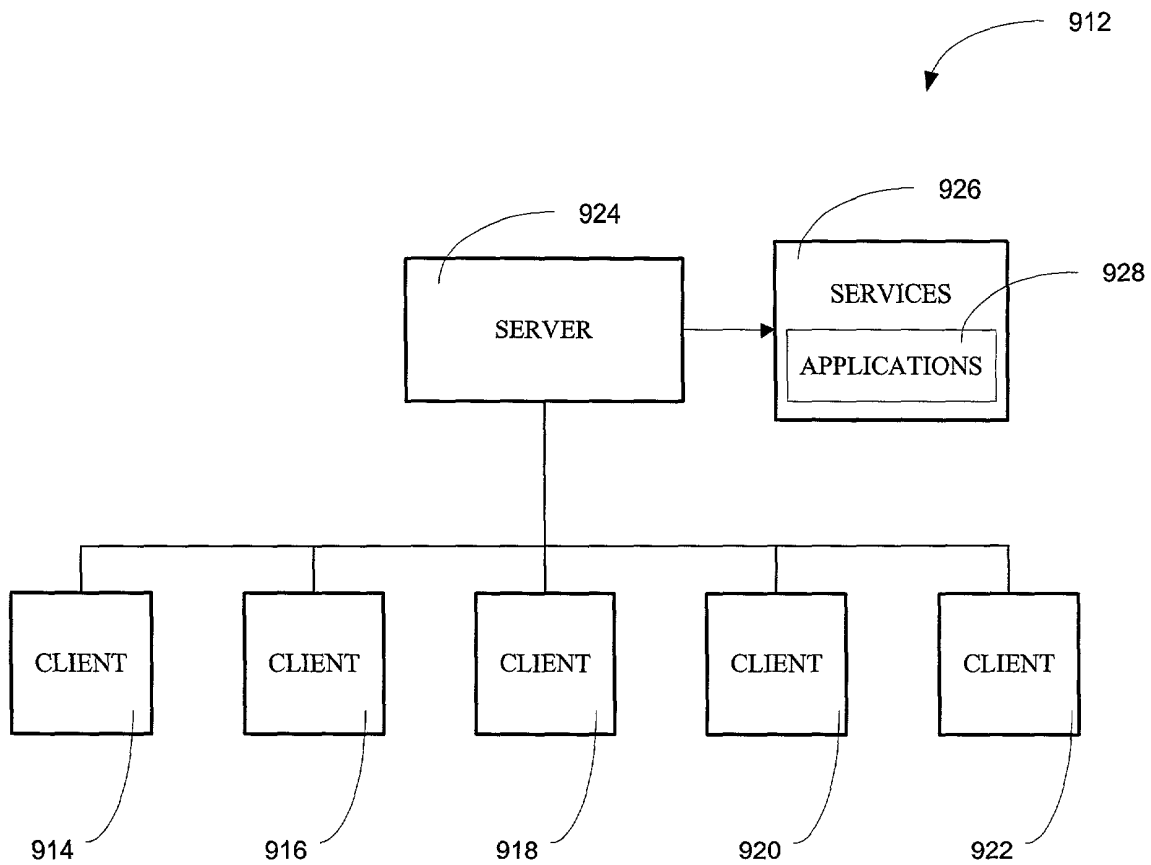
**Fig. 17**



**Fig. 18**

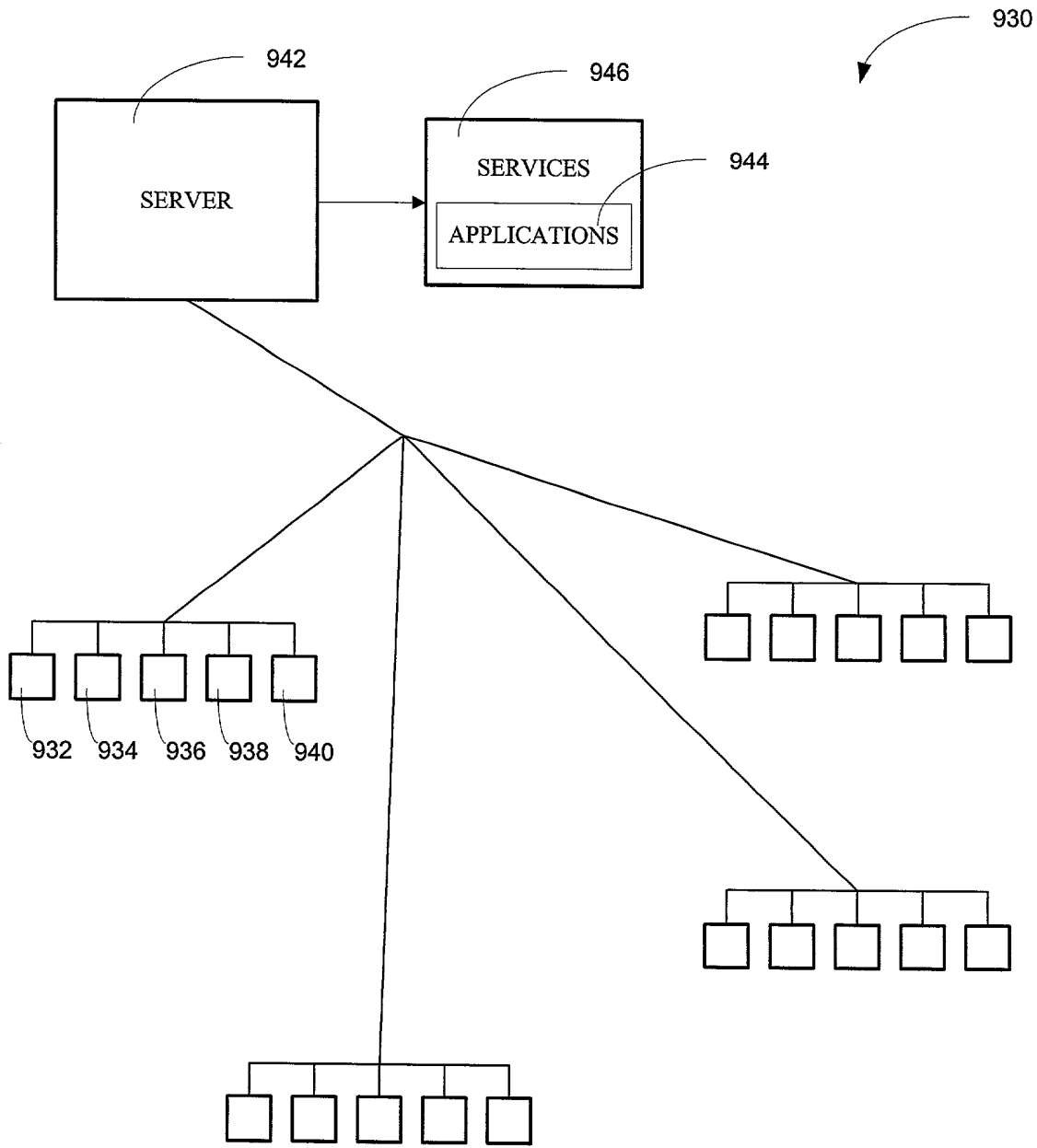


**Fig. 19**



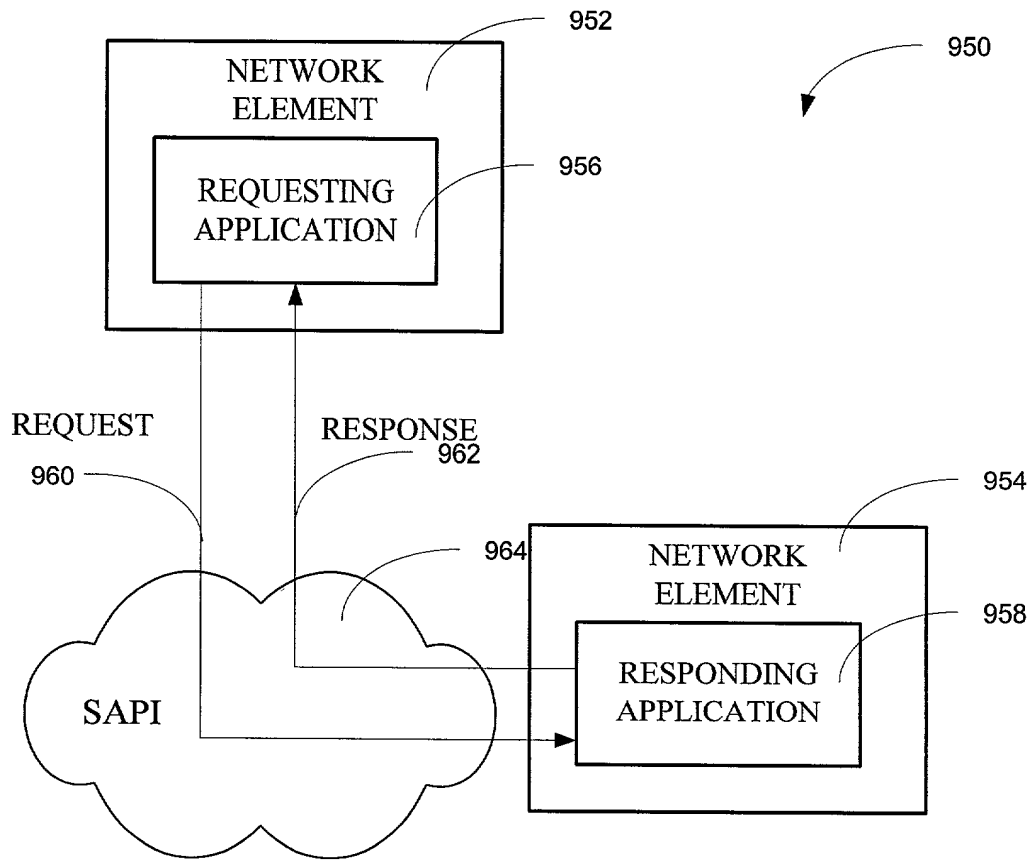
**CLIENT-SERVER  
NETWORK**

**Fig. 20**



WIDE AREA  
NETWORK

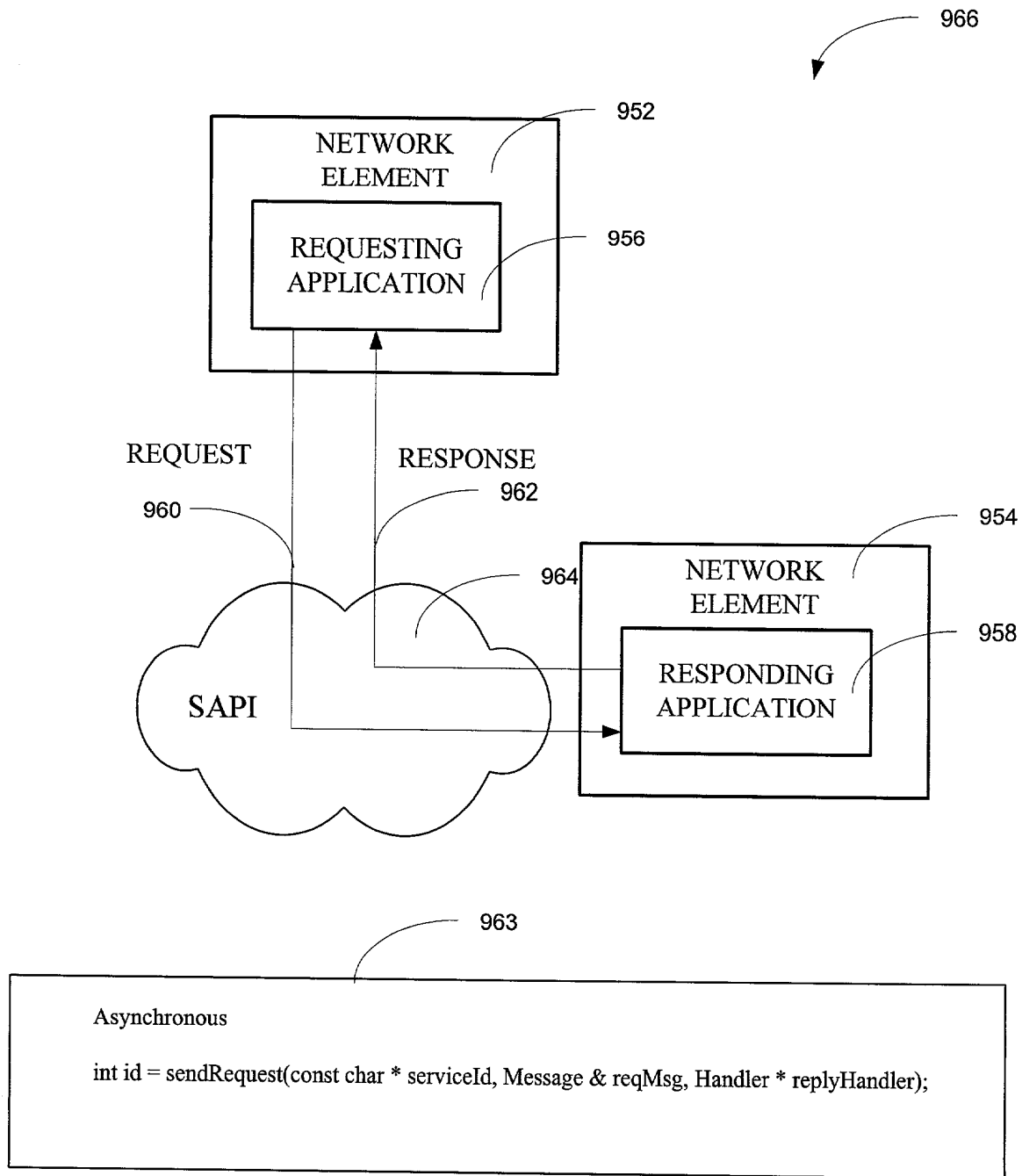
**Fig. 21**



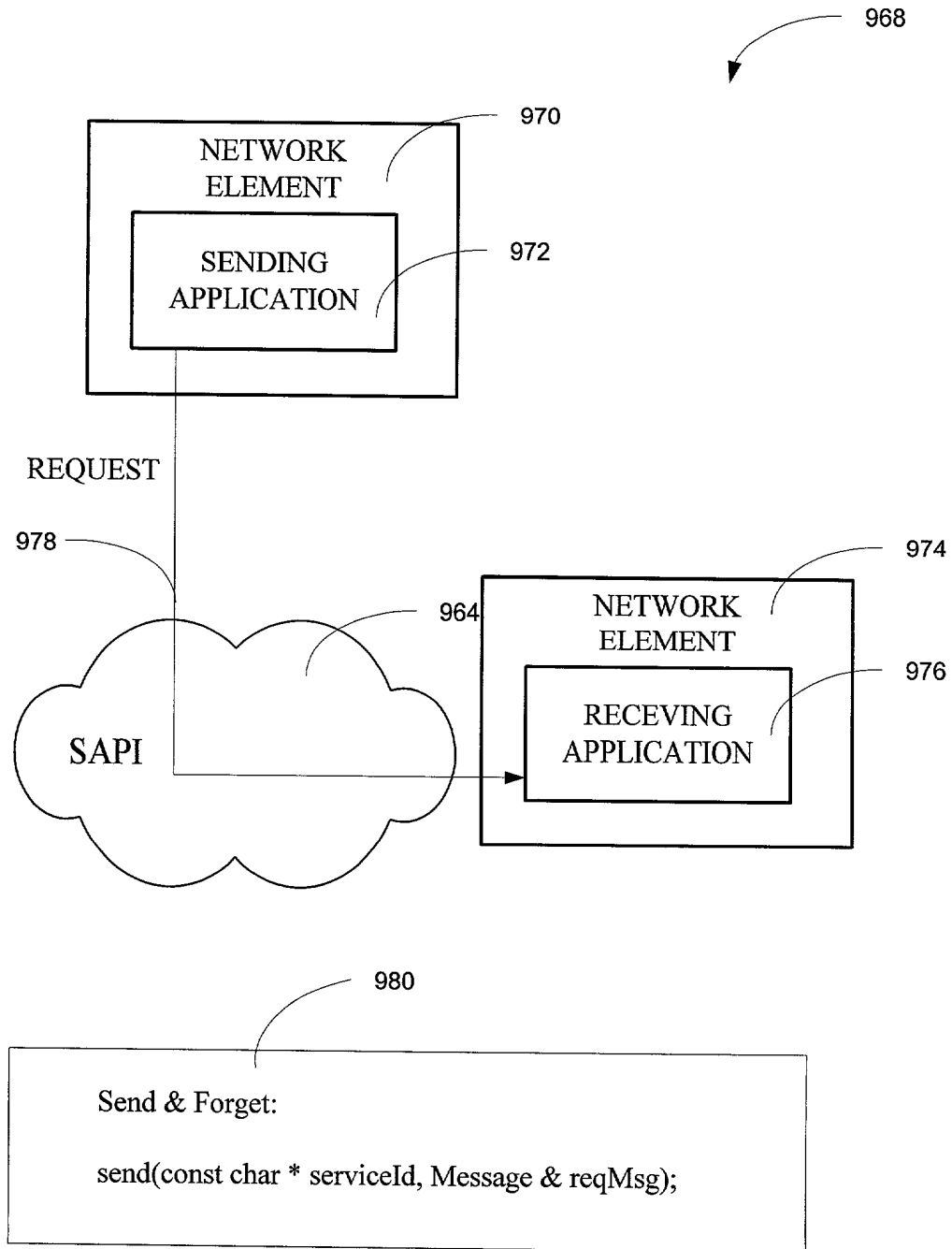
Synchronous mode:

```
Message * reply = call(const char * serviceId, Message & reqMsg);
```

**Fig. 22**

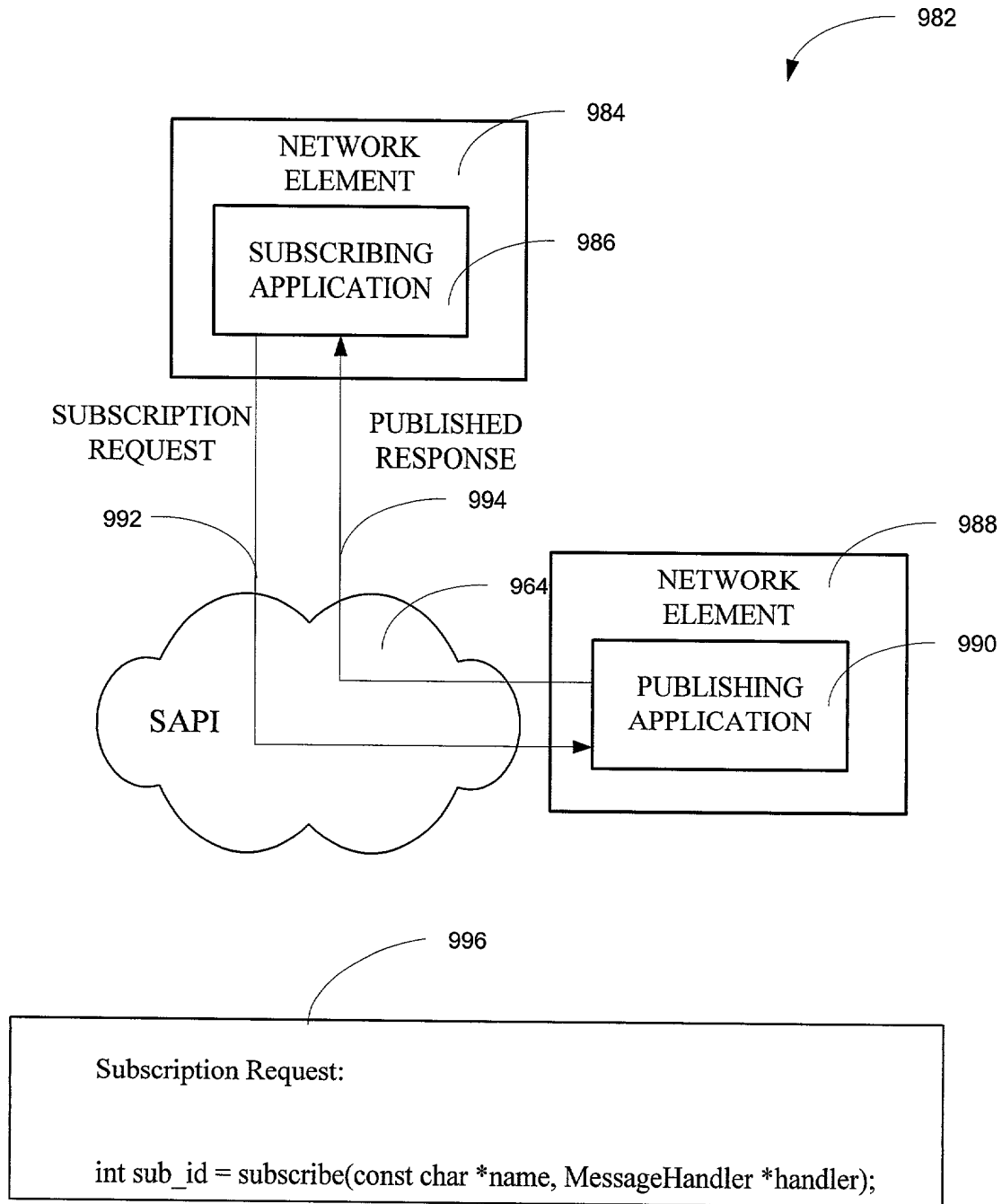


**Fig. 23**



**Fig. 24**





**Fig. 25**